A Pattern Language for Interactive Surfaces in Collaborative Workspaces

Remy, Christian ; Weiss, Malte ; Borchers, Jan

A Pattern Language for Interactive Surfaces in Collaborative Workspaces

Christian Remy
RWTH Aachen University
johannes.christian.remy@rwth-aachen.de

Malte Weiss
RWTH Aachen University
weiss@cs.rwth-aachen.de

Jan Borchers
RWTH Aachen University
borchers@cs.rwth-aachen.de

ABSTRACT
We present a pattern language to capture the experience from proven solutions for collaboration using interactive surfaces. This paper explains the importance of standards and the advantages of pattern languages over traditional guidelines. We describe the pattern structure, the process of creation, and the evaluation of the pattern language.

Author Keywords
Pattern Language, Collaboration, Standards

ACM Classification Keywords
H.5.2 Information Interfaces and Presentation: User Interfaces—*Standardization*

INTRODUCTION
Tabletop interaction is a fast growing area in the research field of human-interaction design. Because of the rich interaction and awareness of direct touch interfaces, multi-touch tables provide a natural environment for collaborative tasks. Research in tabletop interaction has yielded a variety of different solutions and is continuously emerging new ideas [4]. However, it is still hard for designers to find standards to evaluate the design decisions of multi-touch tables against findings from other experts in this field.

We overcome this lack of standards by identifying the recurring problems and their proven solutions, thus creating a comprehensive fund of knowledge in the field of the design of interactive surfaces, avoiding redundancy, and providing a common ground for future work. The nature of pattern languages offers several advantages over traditional guidelines that apply particularly to this domain, guarantees longevity of the proposed patterns, and provides inspiring thoughts for new ideas. Novices familiarize faster with tabletop research but professionals benefit likewise from standards, as they can more easily compare their results with others. Approved review techniques will be used to evaluate the patterns, in particular a writers' workshop [3] with experts as well as novice users from different research fields.

PATTERN LANGUAGES
The concept of pattern languages has been applied to several topics in HCI, e.g., user interface design [5], web design [6], or design of interactive exhibits [1]. Although their format and target audience varies, the definition of patterns has been formulated frequently to achieve a consensus: a pattern captures a proven solution in context to a recurring problem of conflicting forces in interactive design. The notion of context is of utmost importance, since especially the combination of single patterns into a network yields the real capability of a pattern language. This becomes clear by investigating the advantages of pattern languages over traditional guidelines [2]:

- **Lingua franca:** Pattern languages create a common language to enhance communication in both research and applications. It provides a vocabulary for non-experts to take actively part in the design process.
- **Solutions in context:** Design decisions often depend on other, previous considerations [7]. A pattern language is represented as directed acyclic graph (Figure 1). Patterns offer further solutions to following design questions.
- **Readability:** Patterns are written in narrative form and explain the design decision to the reader. They have a consistent format and a clear typography style which makes them easy to skim.
- **Different abstraction levels:** Traditional guidelines focus on one abstraction level. They either capture a broad perspective or center around one particular design aspect. Pattern languages achieve both: top-level patterns handle large-scale problems, while other patterns help to focus on more detailed and concrete issues.
- **Generativity:** Both patterns and the pattern language are generative; the contextual presentation style invites to develop new patterns and alternative solutions.

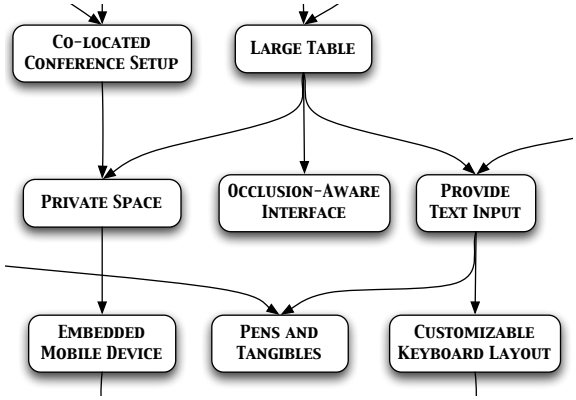

Figure 1. Excerpt from our pattern language.

- **Teaching:** Patterns are validated with evidence from literature, studies, examples, and counter-examples. They explain not only the solution to the reader but they also convey why a particular solution is optimal.

- **Examples:** Every pattern exposes a *rationale* part with examples of the solution, applied in prototypes, tested in user studies, or even used in commercial systems.

PATTERN STRUCTURE

Every pattern language consists of the same elemental parts. Finding an appropriate pattern format specific to tabletop collaboration will be one of our contributions. We will outline the important elements of a pattern language here by exemplifying a tabletop pattern about territoriality, using the pattern format of [1].

- **Name:** The name (written in small capitals) concisely conveys the idea of the pattern to the reader, "PRIVATE SPACE". A ranking is added to indicate the quality of the given solution.

- **Picture:** A picture gives an impression about the idea and is usually represented by a photograph of an example system or a series of pictures like a storyboard or even a very short video, if the patterns are presented in digital form.

- **Context:** A short paragraph integrates the pattern into its application context and names patterns for issues on a larger scale, e.g., "*when using a* LARGE TABLE *in a* CO-LOCATED CONFERENCE SETUP*, consider the use of private space areas*." (Figure 1)

- **Problem statement:** Outlines the problem solved with this pattern. E.g., "*Participants in a collaborative workspace may like to share their documents with others but sometimes they want to keep specific data in private*."

- **Rationale:** The conflicting forces are emphasized and a thorough explanation including successfully applied, working examples from research literature are exposed. In our example, we would name important papers about territoriality, present study results, and highlight the conflicting forces from the problem statement.

- **Solution:** A very short and general solution is presented. E.g., "*Provide a small area of personal space for every user, where she can keep her private data separated from the interaction of the collaborators*."

- **Diagram:** An example sketch or short storyboard exemplifies the application of the solution, without providing a blueprint to copy an existing solution but rather to inspire the reader.

- **References:** Like the context, this section points to other patterns in a lower abstraction level. E.g., "*After implementing private space, consider input of private data using* EMBEDDED MOBILE DEVICES." (Figure 1)

PROCESS

We start the pattern language creation process by reviewing literature about tabletop interaction in collaborative environments and searching for recurring problems with proposed solutions. Accordingly, we identify the conflicting forces that are critical to fulfill the pattern definition. If a problem is resolved in literature and the solution has been applied successfully, the pattern is formulated following the aforementioned structure. The general accepted validation of patterns suggests three evidences in working applications. Furthermore, we intend to collect feedback in multiple review steps. One commonly conducted review process to patterns is a writers' workshop, which includes non-experts as well as professionals, not only from the tabletop domain but also those familiar with pattern languages.

FUTURE WORK

The creation of a pattern language is not a linear process, it follows a cycle of many iterations. Thus, we intend to publish the patterns online and implement various options for feedback and participation. Examples of other pattern languages show how useful this is for the whole community, for example the Yahoo! Design Pattern Library[1]. Our pattern language should be seen as a first step towards a collection of knowledge of interactive surfaces, to strive for consensus and standards and inspire further research.

ACKNOWLEDGEMENTS

This work was funded in part by the German B-IT Foundation.

REFERENCES

1. J. Borchers. *A Pattern Approach to Interaction Design*. John Wiley & Sons, Inc., New York, NY, USA, 2001.

2. A. Dearden and J. Finlay. Pattern Languages in HCI: A Critical Review. *Human-Computer Interaction*, pages 49–102, Mar. 2006.

3. R. P. Gabriel. *Writers' Workshops & the Work of Making Things: Patterns, Poetry...* Pearson Education, 1st edition, June 2002.

4. T. Grossman and D. Wigdor. Going Deeper: a Taxonomy of 3D on the Tabletop. In *TABLETOP '07*, pages 137–144, Oct. 2007.

5. J. Tidwell. *Designing Interfaces: Patterns for Effective Interaction Design*. O'Reilly Media, Inc., Nov. 2005.

6. D. K. van Duyne, J. A. Landay, and J. I. Hong. *The Design of Sites: Patterns for Creating Winning Web Sites*. Prentice Hall PTR, 2nd edition, Dec. 2006.

7. J. Wallace and S. Scott. Contextual Design Considerations for Co-located, Collaborative Tables. In *TABLETOP '08*, pages 57–64, Oct. 2008.

[1] http://developer.yahoo.com/ypatterns/